

# Linguagens de Programação 2016/2017

## Gramáticas

1. Desenhe as **árvores abstractas de sintaxe** para as seguintes expressões (considere *sqrt* um operador com um único argumento):

(a)  $a * b + c$

(b)  $a * (b + c)$

(c)  $a * b + c * d$

(d)  $a * (b + c) * d$

(e)  $(b/2 + \text{sqrt}((b/2) * (b/2) - a * c))/a$

2. Re-escreva as mesmas expressões em notação **prefixa**.
3. Re-escreva as mesmas expressões em notação **posfixa**.
4. Em cada um dos seguintes casos, as regras BNF descrevem uma construção que consiste numa lista de elementos. Descreva a lista indicando:
  - o que é cada elemento
  - nº mínimo de elementos
  - o que é e se separa ou termina cada elemento
  - o tipo de associatividade

(a)  $\langle \text{name-list} \rangle \rightarrow \langle \text{name} \rangle \mid \langle \text{name} \rangle, \langle \text{name-list} \rangle$

(b)  $\langle \text{field-list} \rangle \rightarrow \langle \text{field} \rangle; \mid \langle \text{field} \rangle; \langle \text{field-list} \rangle$

(c)  $\langle \text{statement-list} \rangle \rightarrow \langle \text{empty} \rangle \mid \langle \text{statement} \rangle; \langle \text{statement-list} \rangle$

(d)  $\langle \text{term} \rangle \rightarrow \langle \text{term} \rangle * \langle \text{factor} \rangle \mid \langle \text{factor} \rangle$

(e)  $\langle \text{variables} \rangle \rightarrow \langle \text{empty} \rangle \mid \text{var } \langle \text{var-decls} \rangle$   
 $\langle \text{var-decls} \rangle \rightarrow \langle \text{name-list} \rangle : \langle \text{type} \rangle; \mid \langle \text{name-list} \rangle : \langle \text{type} \rangle; \langle \text{var-decls} \rangle$

(f)  $\langle \text{constants} \rangle \rightarrow \langle \text{empty} \rangle \mid \langle \text{c-decls} \rangle$   
 $\langle \text{c-decls} \rangle \rightarrow \text{const } \langle \text{name} \rangle = \langle \text{constant} \rangle; \mid \langle \text{c-decls} \rangle \langle \text{name} \rangle = \langle \text{constant} \rangle;$

5. Defina uma gramática não ambígua para expressões lógicas; os valores lógicos são representados por **false** e **true** e as expressões podem conter os operadores disjunção (**||**), conjunção (**&&**) e negação (**not**). Tenha em conta as prioridades relativas usuais destes operadores e que as expressões devem ser avaliadas da esquerda para a direita (onde aplicável).
- (a) Acrescente parêntesis à linguagem.
  - (b) Acrescente o operador de implicação (**=>**). Lembre-se de que o operador de implicação associa à direita.
6. Defina uma gramática não ambígua para as expressões matemáticas com números naturais e as operações soma, subtração, produto e divisão. Tenha em conta as prioridades e a associatividade usuais das operações matemáticas.
- (a) Acrescente o operador de potenciação (**^**) à linguagem (este operador associa à direita).
  - (b) Acrescente parêntesis.
7. Se uma gramática tiver as seguintes duas produções, ela é ambígua
- ```
S ::= if E then S
S ::= if E then S else S
```
- (a) Justifique porquê.
  - (b) Escreva uma gramática não ambígua que gere as mesmas condições e que ligue um **else** com o **if** não ligado que lhe está mais próximo.